基于分层强化学习的自动驾驶车辆掉头问题研究

曹 洁，邵紫旋，侯 亮†
(兰州理工大学 计算机与通信学院，兰州 730050)

摘 要：调头任务是自动驾驶研究的内容之一，大多数在城市规范道路下的方案无法在非规范道路上实施。针对这一问题文中建立了一种车辆掉头动力学模型，并设计了一种多尺度卷积神经网络提取特征图作为智能体的输入。另外文中还针对调头任务中的稀疏奖励问题，结合分层强化学习和近端策略优化算法提出了分层近端策略优化算法，在简单和复杂场景的实验中，该算法相比于其他算法能够更快的学习到策略，并且具有更高的掉头成功率。

关键词：分层强化学习；汽车掉头；稀疏奖励；近端策略优化
中图分类号：TP181 doi: 10.19734/j.issn.1001-3695.2022.03.0127

Research on autonomous vehicle u-turn problem based on hierarchical reinforcement learning

Cao Jie, Shao Zixuan, Hou Liang†
(*Dept. of Computer & Communication*, *Lanzhou University of Technology*, *Lanzhou* 730050, *China*)

Abstract: The U-turn task is one of the contents of autonomous driving research, and most of the solutions under the standard roads in cities cannot be implemented on non-standard roads. Aiming at solving this problem, this paper establishes a vehicle U-turn dynamical model and designs a multi-scale convolutional neural network to extract feature maps as the input of the agent. In addition, for the sparse reward problem in the U-turn task, this paper proposes a hierarchical proximal policy optimization algorithm that combines hierarchical reinforcement learning and proximal policy optimization algorithm. In experiments with simple and complex scenarios, this algorithm learns policies faster and has a higher success rate of U-turn compared to other algorithms.

Key words: hierarchical reinforcement learning; car u-turn; sparse rewards; proximal policy optimization

0 引言

随着经济不断发展，人们对自动驾驶车辆的要求也逐步提高。现有的自动驾驶车辆已经能够在城市道路和高速公路上行驶，它通过地图数据与全球定位系统(global positioning system，GPS)定位信号或者车载摄像头来获取车辆位置，通过识别道路上的路面标记、交通标志以及交通信号灯来作出正确的决策。但在一些地下停车场、小区车道等路况复杂的空间场景，GPS 信号较弱，同时缺乏路面标记以及交通辅助信息，自动驾驶车辆往往难以应对此类场景。

传统的自动驾驶系统[1~3]在设计的过程中被分解为多个子系统，通过子系统之间的相互配合来完成自动驾驶任务，并在一些复杂场景中设计大量的子模块辅助车辆进行自动驾驶，这样的设计使得自动驾驶技术非常复杂，维护成本高昂。近些年，人工智能技术[4~6]发展迅猛，尤其是强化学习[7~13]展现出了巨大的潜力。强化学习分为基于模型的强化学习方法[7,8]和无模型的强化学习方法[9~12]。它是一种学习、预测、决策的方法框架，也是一种致力于实现通用智能解决复杂问题的方式。但是传统的强化学习方法在一些奖励稀疏[14~17]问题上表现较差，针对该问题，一些研究人员提出使用分层强化学习[17~20]的方法解决。

强化学习在自动驾驶领域也有大量的应用[21~25]，在驾驶车辆的过程中，驾驶员需要时刻注意车辆周围的环境情况，不断根据周围环境的变化作出决策，而深度强化学习技术能解决端到端的感知与决策问题，越来越多的学者开始将深度强化学习应用在自动驾驶领域。

Li 等人[21]为寻找具有风险意识且能够使得风险最小的自动驾驶决策策略，提出了一种基于深度强化学习的变道决策框架。Peng 等人[22]通过给十字路口的一部分自动驾驶车辆设计一个利他的奖励功能，来提高整个交叉路口的通行效率。WANG 等人[23]基于强化学习的端到端自动驾驶模型提出了一种异步监督学习方法，以解决在真实环境中训练该模型的初始性能较差的问题。Kim 等人[24]利用强化学习对现有的自动驾驶模型进行了修正和改进，提出了一种自动驾驶预测模型，减少了训练时间，并提高了驾驶表现。Kendall 等人[26]首次演示了深度强化学习在自动驾驶中的应用，他们的模型能够使用单一的单眼图像作为输入，在少量的训练集中学习车道跟随策略。相比传统的自动驾驶技术，深度强化学习技术不用设计繁多的任务模块，可以模拟人的驾驶行为，从“端到端”解决自动驾驶问题。

但自动驾驶车辆应当能够应对生活中出现的各类场景，能够在各种情况下完成自动驾驶任务。除了高速公路以及城市道路，自动驾驶车辆也应当能够在一些不规范道路，比如小区车道，停车场车道等道路上进行自动驾驶。目前城市道路场景(比如提高十字路口通行效率、超车、跟车等行为)以及高速公路场景的自动驾驶已经存在比较多的研究，然而在其他场景下自动驾驶任务仍需要作出一些工作。比如在此类道路进行一些掉头、转弯等行为，当在此类地区进行自动驾驶时，可以使用车辆传感器对道路环境进行观测，然后通过车载计算机计算出最佳行进路线，最后车辆根据车载计算机规划出来的路线完成自动驾驶任务。

本工作使用深度强化学习技术，针对一些缺乏自动驾驶辅助信息的场景，建立了马尔可夫决策过程(Markov decision process, MDP)模型，提出了一种自动驾驶车辆在不规范车道

收稿日期：2022-03-09；**修回日期：**2022-05-13

作者简介：曹洁(1966-)，女，安徽宿州人，教授，博导，硕士，主要研究方向为人工智能；邵紫旋(1996-)，男，甘肃平凉人，硕士研究生，主要研究方向为强化学习、智能交通系统；侯亮(1976-)，男(通信作者)，甘肃兰州人，博士研究生，主要研究方向为智能信息处理，智能交通(zxuanshao@163.com).

下的掉头方法。考虑到车载摄像头难以应对全天候工作，视频图像信息容易受到对抗样本的攻击等问题[27,28]，因而采用激光雷达传感器进行采集信息作为输入。

整体上这篇工作主要的贡献点在于：

a) 一个在不规则车道场景下的车辆掉头 MDP 模型在这篇论文中提出，用作不规范道路下的自动驾驶任务。

b) 一种多尺度融合卷积神经网络被用作提取状态值特征的任务，取得了很好的效果。

c) 一种针对车辆调头任务奖励问题提出的分层近端策略优化算法(hierarchical proximal policy optimization，HPPO)，其效果在简单与复杂场景中得到验证。

1 强化学习

为了更好的解决车辆掉头问题，先将其抽象为马尔可夫决策过程，然后使用强化学习的方法来解决这一问题。MDP 包含几个重要的元素：(S,A,R,γ)，其中 S 代表环境状态，A 代表智能体的动作，R 代表环境的回报，一次完整的状态转换可以表示为：t 时刻的环境状态为 s_t ,在智能体执行动作 a_t 后环境状态转变为 s_{t+1}，同时环境反馈给智能体 r_t 的奖励，这一系列状态、动作、奖励的轨迹定义为 τ，如式(1)所示。

$$\tau=(s_0,a_0,r_0,s_1,a_1,r_1,\cdots,s_{|T|},a_{|T|},r_{|T|}) \tag{1}$$

$|T|$ 代表决策序列的长度，强化学习的最终目标是通过智能体与环境不断交互得到最大累计奖励 r_{total}，如式(2)所示。

$$r_{total}=\sum_{t=0}^{|T|}r_t \tag{2}$$

在智能体与环境不断交互的过程中，为了获取最高奖励，智能体通过学习选取价值最优的策略(动作)，可通过如 Q 价值函数表示，如式(3)所示。

$$Q_\pi(s,a)=E_\pi\{G_t\,|\,S_t=s,A_t=a\} \tag{3}$$

其中，G_t 表示 t 时刻的状态到达最终状态的累计奖励。Q 值用于评判动作的好坏，状态的好坏使用 V 值来评判，并且 V 价值函数可基于 Q 价值函数值来计算，如式(4)所示。

$$V_\pi(s)=\sum_{a\in A}\pi(a\,|\,s)Q_\pi(s,a) \tag{4}$$

其中，$\pi(a|s)$ 表示智能体的策略，即在状态 s 下选择动作 a 的概率。

表演家-评论家(Actor-Critic)算法融合了基于价值的方法与基于策略的方法，它使用表演家(Actor)网络学习策略，又通过评论家(Critic)网络估计的价值函数进行策略更新，它解决了基于策略的方法的高方差问题，并且更容易处理连续行为。它是一种近似的策略梯度，其梯度计算如式(5)所示。

$$\nabla_\theta J(\theta)\approx\mathbb{E}_{\pi\theta}[\nabla_\theta\log\pi_\theta(s,a)Q_\omega(s,a)] \tag{5}$$

其中，ω 表示评论家网络更新的动作值函数，θ 表示表演家网络以评论家网络所指导的方向更新策略参数。

近端策略优化算法(Proximal Policy Optimization，PPO)算法是基于表演家-评论家框架的算法，它是在基于置信域的策略优化(Trust region policy optimization, TRPO)算法的基础上进行了改进，优化了更新参数的方式。近端策略优化算法采用阶段代理目标函数来控制策略的更新，它将新旧策略的比值限制在一个范围内，通过控制这个范围的大小来限制更新的幅度。近端策略优化算法的目标函数如式(6)所示。

$$J^{CLIP}(\tilde{\theta})=\mathbb{E}_{s_t\sim\rho_{\pi\theta},a_t\sim\pi_\theta}[\min(k_t(\tilde{\theta}),clip(k_t(\tilde{\theta}),1-\epsilon,1+\epsilon))\hat{A}_\theta(s,a)] \tag{6}$$

其中，ϵ 是用于度量新策略与老策略之间偏差程度的超参数，$clip(k_t(\tilde{\theta}),1-\varepsilon,1+\varepsilon)$ 将重要性采样权重限制在 $(1-\epsilon,1+\epsilon)$ 的范围内。$k_t(\tilde{\theta})$ 表示新旧策略的比值，如式(7)所示。

$$k_t(\tilde{\theta})=\frac{\pi_\theta(a,s)}{\pi_{\theta^-}(a,s)} \tag{7}$$

但是单层结构的强化学习算法在应对一些奖励稀疏的问题时，常常难以发挥出其优越的性能。分层强化学习通过将问题分解为一组短期子问题来加速稀疏奖励任务中的学习。分层表演家-评论家(hierarchical actor-critic, HAC)算法是第一个成功地在具有连续状态和动作空间的任务中并行学习三级层次结构的框架，它通过设计的三种转换，并行的训练多个层级。文章通过在马尔可夫决策过程中增加了一组目标 G，构建了通用马尔可夫决策过程(universal markov decision process, UMDP)，所以通用马尔可夫决策过程包含的元素为：(S,G,A,R,γ)，其中 G 是目标集合，它的 Q 值与 V 值计算如式(8)与式(9)所示。

$$Q_\pi(s,g,a)=E_\pi[\sum_{n=0}^{\infty}\gamma^n R_{t+n+1}|s_t=s,g_t=g,a_t=a] \tag{8}$$

$$V_\pi(s,g)=E_\pi[\sum_{n=0}^{\infty}\gamma^n R_{t+n+1}|s_t=s,g_t=g] \tag{9}$$

其中，$g\in G$ 是整个回合的目标，每一层级的状态、动作集合以及最底层的动作空间均与原始空间相同，低一层智能体通过 $s\times g\to A$ 来最大化价值函数。

分层近端策略优化算法采用分层表演家-评论家算法的框架，在分层机制的基础上，利用近端策略优化算法来更新表演家网络和评论家网络。

2 车辆掉头动态模型建立

由于目前的自动驾驶算法训练平台，如开放赛车模拟器(the open racing car simulator，TORCS), Air Sim, Carla 等都无法自定义场景，且难以二次开发。所以解决自动驾驶车辆的掉头问题，首先要针对场景建立模型与仿真环境，最后选用合适的强化学习算法进行求解。

模型选用车辆的位置与转弯角度作为状态，选择车辆的转弯角度作为动作，车辆每一时刻的位置可根据上一时刻的位置计算得到，如式(10)所示。

$$\begin{cases}x_{t+1}=x_t+\int v\sin(\theta+\Delta\theta)dt\\ y_{t+1}=y_t+\int v\cos(\theta+\Delta\theta)dt\end{cases} \tag{10}$$

$\Delta\theta$ 表示动作执行后，汽车转弯角度的变化量，最后构建奖励函数 R 来建立车辆动力模型，奖励函数如式(11)所示。

$$R=\begin{cases}r=0(\text{车辆正常行驶})\\ r=10-\alpha A_{count}(\text{车辆成功掉头})\\ r=-10(\text{车辆触碰边界})\end{cases} \tag{11}$$

奖励函数设计的好坏直接影响着算法的收敛与否以及算法的收敛速度。由于仿真车辆在掉头过程中所做的动作难以判定好坏，所以将仿真车辆行驶时刻的奖励设置为 0。当仿真车辆触碰边界时，给智能体一个较大的负奖励，促使其尽量避免触碰边界；当仿真车辆成功掉头时，给它一个正奖励，并减去掉头过程中使用的动作总数 A_{count} 与参数 α 的乘积，经过反复实验，最终取 α 为 0.1。

将车辆的位置与转弯角度信息进行卷积操作后输入特征提取网络，然后将状态特征输入智能体，智能体经过处理后输出动作信息给环境，然后环境给智能体反馈奖励信号，模型原理图如图 1 所示。

本研究针对所建立的模型构建了虚拟仿真环境，地图的大小设置为 400×600，以左下角为原点，在这张地图中，绿色部分是不可行驶区域，灰色部分是可行驶区域，黑色直线表示场景边界。

设定车辆在掉头过程中的速度是恒定的，车辆在掉头过程中不能在不可行驶区域行驶。车辆掉头仿真环境如图 2 所示。

根据仿真环境的大小,形状和车辆的动态特性,和一个规则的车辆不能碰撞仿真环境的边缘等因素;将奖励值的定义规则如下：

a) 当 $0+L<y<300-L$，并且 $x<100+L$，$x>300-L$ 时，表示车辆行驶到了不可行驶区域，此时 $r=-10$，学习过程结束并重新开始。

b) 当 $300+L<y<600-L$，并且 $x<0+L$ 或 $x>400-L$ 时，表示车辆撞到了地图的左右边界，此时 $r=-10$，学习过程结束并重新开始。

c) 当 $0+L<x<400-L$，并且 $y<0+L$ 或 $y>600-L$ 时，表示车辆撞到了地图的上下边界，此时 $r=-10$，学习过程结束并重新开始。

d) 当 $0+L<y<100-L$，并且 $100+L<x<300-L$ 时，车辆到达目的地，$r=10-\alpha A_{count}$。

e) 其他情况，仿真车辆被认为在模拟场景中行驶，$r=0$。

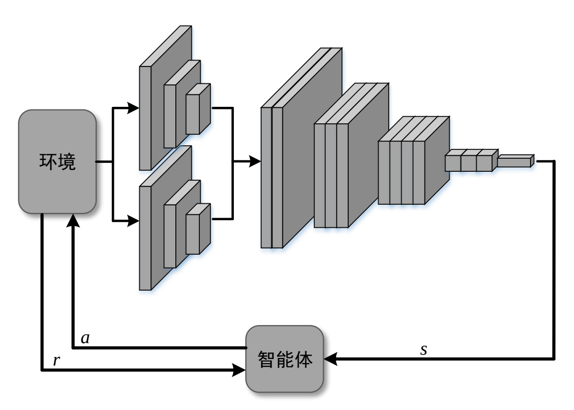


图 1　模型原理图

Fig. 1　Model schematic diagram

在建立第一个环境时，由于车辆可行驶区域较大，在训练过程中，仿真车辆不用倒车也可以实现掉头行为，所以为了增加实验难度，让自动驾驶车辆能够适应更多的复杂环境，第二个实验缩小了仿真车辆用于转弯掉头的可行驶区域，此时仿真车辆必须在转弯过程中进行倒车才能完成掉头任务，增加掉头难度后的仿真环境示意图如图 3 所示。

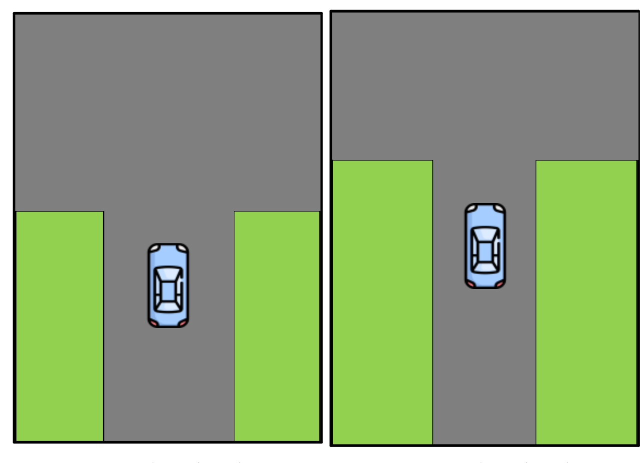


图 2　仿真环境示意图

Fig. 2　Schematic diagram of simulation environment

图 3　仿真环境示意图

Fig. 3　Schematic diagram of simulation environment

环境改进后，此时奖励值的定义规则如下：

a) 当 $0+L<y<400-L$，并且 $x<100+L$，$x>300-L$ 时，表示车辆行驶到了不可行驶区域，此时 $r=-10$，学习过程结束并重新开始。

b) 当 $400+L<y<600-L$，并且 $x<0+L$ 或 $x>400-L$ 时，表示车辆撞到了地图的左右边界，此时 $r=-10$，学习过程结束并重新开始。

c) 当 $0+L<x<400-L$，并且 $y<0+L$ 或 $y>600-L$，表示车辆撞到了地图的上下边界，此时 $r=-10$，学习过程结束并重新开始。

d) 当 $0+L<y<100-L$，并且 $100+L<x<300-L$，车辆到达目的地，$r=10-\alpha A_{count}$。

e) 其他情况，车辆被认为在模拟场景中行驶，$r=0$。

在强化学习当中，奖励函数对智能体的训练至关重要，其承担了类似于监督学习中数据标签的作用。一方面，由于刚开始训练时，智能体采用随机策略，导致智能体获取奖励难度较大，所以刚开始训练智能体时得到的奖励相对稀疏；另一方面，稀疏奖励广泛存在于一些强化学习任务之中。比如在机械臂抓取任务中，机械臂要完成一系列复杂的动作才能成功抓取目标，获得最终奖励，中间任何一个动作导致实验失败都无法获取最终奖励，但除去导致机械臂抓取任务失败的少部分动作外，该过程中的其他动作很难判定其好坏，也很难给与这些动作确定的奖励；在飞行器导航任务中，只有当飞行器成功到达指定位置或撞毁在障碍物上时才能获得最终奖励或惩罚，飞行过程中飞行器所做的一系列调整飞行姿势的动作都很难设定奖励；还有围棋等强化学习任务都属于稀疏奖励问题，在使用深度强化学习解决实际问题时经常面临着该问题，它会大大降低算法的迭代速度，甚至会导致算法难以收敛。仿真环境中的奖励示意图如图 4 所示。

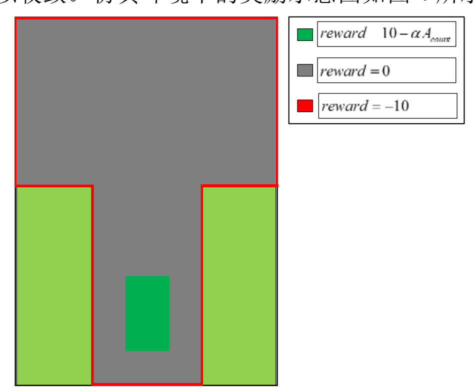


图 4 仿真环境的奖励示意图

Fig. 4　The reward schematic of the simulation environment

在仿真环境中，浅绿色部分代表不可行使区域，灰色部分代表可行驶区域。在可行驶区域内部，仿真车辆驶入深绿色的方框代表到达目标区域，此时获得奖励，红色的线代表仿真环境的边界，当仿真车辆在行驶过程中碰到红色的线代表此回合训练失败，此时获得惩罚；在其他情况下，即仿真车辆行驶在灰色可行驶区域，未进入目标区域也未触碰仿真环境边界，奖励为零。由奖励示意图可观察到，没有奖励或惩罚的状态的数量要远远超过有奖励的状态的数量，在实验中，有确定奖励的状态非常稀疏。

3　分层近端策略优化算法

很多传统的强化学习算法采用同策略的方式一边与环境交互，一边进行学习，这样大大降低了智能体的学习速度，近端策略优化算法通过重要性采样将同策略改进为异策略，提高了智能体的学习速度，重要性采样公式如式(12)所示。

$$E_{x\sim p}[f(x)]=E_{x\sim q}[f(x)\frac{p(x)}{q(x)}] \tag{12}$$

通过智能体与环境的交互得到可以得到轨迹 τ，然后使用评论家网络计算出优势函数 G，用于评判所选动作相比其他动作的优势，优势函数如式(13)所示。

$$G_t=r_t+\gamma r_{t+1}+\gamma^2 r_{t+1}+\cdots+\gamma^N r_{t+N}-vs_{(t)} \tag{13}$$

经过反复实验，将 γ 设置为 0.9。在实验中，智能体的网络与优势网络除了输出层，其他部分都使用相同的神经网络结构，每个步骤的回报可按式(14)计算：

$$R_t=G_t+v(s_t) \tag{14}$$

有了优势函数，就可以使用梯度搜索来调整网络参数 θ，

搜索的目的是将式(15)目标函数 $J(\theta)$ 最大化，

$$J(\theta)=\min(p_t(\theta)G_t, clip(p_t(\theta),1-\epsilon,1+\epsilon)G_t) \tag{15}$$

近端策略优化算法网络结构如图 5 所示。

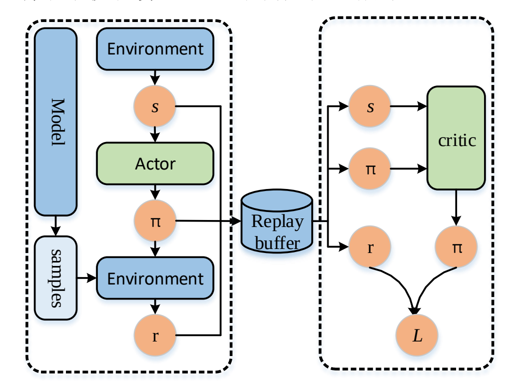


图 5　PPO 算法网络结构图

Fig. 5　PPO algorithm network structure diagram

具有层次结构的智能体能够将强化学习问题分解成更小的子问题，具有加速学习的潜力，所以使用分层强化学习的思想来解决该问题。将智能体的控制分为高层与低层，高层智能体进行低层智能体学习目标的设定，它对多个时间步 t 执行一次决策，高层智能体学习的动力是外部的稀疏奖励；低层智能体通过学习完成高层智能体给定的目标，在每一时间步 t 作出决策，低层智能体的驱动力是内部奖励，如图 6 所示。

智能体的高层与低层都是由近端策略优化算法组成，高层智能体观测原始状态，通过计算价值函数 $Q_2=(s_t,g_t;\theta_2)$ 来最大化外部奖励，低层智能体中的表演家网络接受状态与当前目标，通过计算价值函数 $Q_1=(s_t,a_t;\theta_1,g_t)$ 来求解预测目标，当且仅当目标达成时，评论家网络才会给出正向激励。

当每一回合结束，或目标 g 达成时，低层智能体表演家网络停止，然后高层智能体选择一个新的 g ，然后重复该过程。使用深度学习的框架为高层智能体与低层智能体学习策略，使用式(16)来估计低层智能体的 Q 函数。

$$Q_1^*(s,a,g)=\max_{\pi_{ag}} E[\sum_{t'=t}^{\infty}\gamma^{t'-t}r_{t'} \mid s_t=s,a_t=a,g_t=g,\pi_{ag}] \tag{16}$$

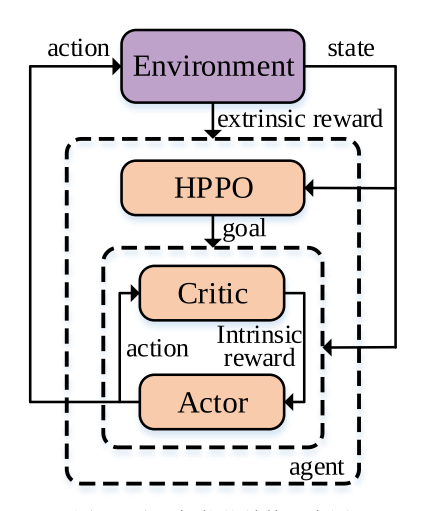


图 6　分层智能体结构示意图

Fig. 6　Schematic diagram of hierarchical agent structure

在高层智能体的策略里面， g 是智能体在状态 s 与策略 π_{ag} 下的目标，同样用式(17)来估计高层智能体的 Q 函数。

$$Q_2^*(s,g)= \max_{\pi_g} E[\sum_{t'=t}^{t+N} f_{t'}+\gamma \max_{g'} Q_2^*(s_{t+N},g') \mid s_t=s,g_t=g,\pi_g] \tag{17}$$

N 代表低层智能体到达当前目标所使用的时间步，g' 表示在状态 s_{t+N} 时智能体的目标， π_g 是当前策略的目标。

使用参数为 θ 的非线性函数近似表示 $Q^*=(s,g)\approx Q(s,g;\theta)$ ， Q_1 、 Q_2 可以通过最小化其损失函数 $L_1(\theta_1)$ 与 $L_2(\theta_2)$ 得到， Q_1 的损失函数可以使用式(18)表示：

$$L_1(\theta_1,i)=E_{(s,a,g,r,s')}\sim(y_{1,i}-Q_1(s,a;\theta_{1,i},g))^2 \tag{18}$$

其中 i 代表训练迭代数， $\theta_{1,i-1}$ 在上一次迭代中保持固定。 $y_{1,i}$ 表示通过上一状态以及其目标得到的 Q 值。损失函数 L_2 的原理与上式相同。

在训练过程中，智能体首先与环境进行交互采集轨迹数据，并将交互得到的轨迹数据存储在经验池中，等存储了足够的数据后，智能体开始在经验池中随机抽取一定量的数据一边交互一边学习，表演家网络进行策略的更新，评论家网络进行价值的更新，更新过程如图 7 所示。

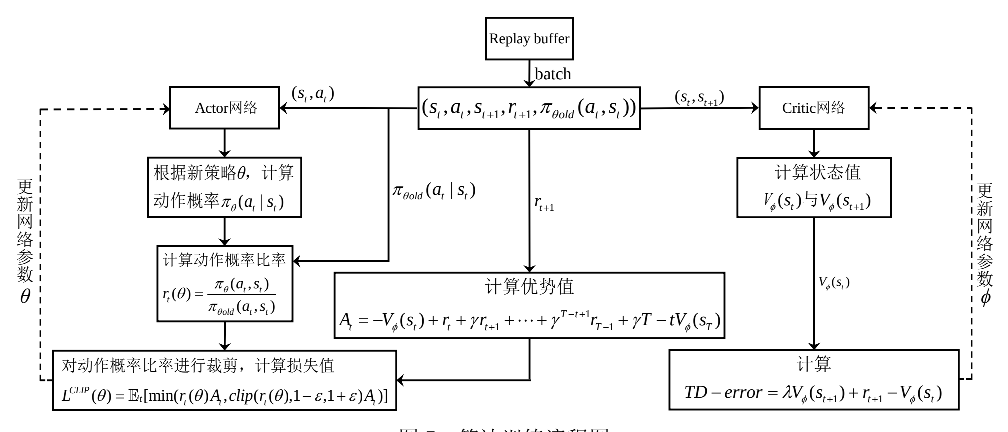


图 7　算法训练流程图

Fig. 7　Schematic diagram of hierarchical agent structure

4　实验

在真实环境中，当可掉头区域较大时，车辆通过在前进过程中多次旋转方向盘，调整车身的位置，从而完成掉头任务，这对应场景一。但在可掉头区域较小时，车辆需要通过增加倒车行为来完成转弯过程，这对应场景二。

4.1　参数设定

在实际情况中，由于车载摄像机的一部分局限性，实验选择使用激光雷达来探测掉头过程中车辆在环境中的位置以及车辆在环境中的姿势。按照实际的比例，在实验的仿真环境中建立一个车宽 40，车长 60 的仿真车辆。然后在仿真车辆的正前方、正后方、正左方、正右方设置四个仿真雷达，通过它们探测的数据计算车辆在仿真环境的坐标与车辆的转弯角度，在实验中，使用这两个量作为智能体的状态。一般的小型车辆最大转弯角度都在 45° 左右，将汽车转弯角度离散化为 5 个选项，每个选项为 18° ，使用其作为智能体的动作。

在场景一中，由于可用于仿真车辆掉头的车辆可行驶区域较大，所以车辆能够在不倒车的情况下使用转弯动作完成

掉头。但在场景二中，由于实验缩小了仿真车辆在转弯过程中的车辆可行驶区域，车辆无法仅通过前进完成掉头任务，所以针对第二个场景，实验二在实验一的基础上又增加了五个倒车动作，分别对应前五个角度的反方向。

在仿真环境中，实验通过仿真车辆的雷达获得车辆位置，以及车辆的旋转角度，将其作为算法的输入，然后输出车辆在下一时刻的旋转角度，在反复进行多次实验后，取 γ 为 0.9，此时算法能获得相对较高的奖励。实验中模型的参数如表 1 所示。表 1 中，仿真车辆在仿真环境中的坐标用 (x,y) 表示，它包含在在仿真环境 D_{xy} 中，仿真车辆的在仿真环境中的车身姿势用 θ 表示，它的范围包含在前进动作空间 A_s 与倒车动作空间 A_b 中。

详细的参数设置如表 1 所示。

表 1　MDP 参数表

Tab. 1　MDP parameter table

MDP	仿真环境
S	$(x,y)\in D_{xy}$; $\theta\in A_s,A_b$
A	场景一：$\theta\in A_s$；场景二：$\theta\in A_b$
R	$r=0$ 仿真车辆未触碰边界未成功掉头；$r=10-\alpha A_{count}$ 仿真车辆成功掉头；$r=-10$ 仿真车辆触碰边界
γ	0.9

4.2　实验

实验采用了 HAC 算法、PPO 算法、AC 算法、DQN 算法与文中提出的 HPPO 算法来测试车辆在初始角度不同时能否训练有效的转弯策略，实验结果如图 8 所示，横坐标为训练回合数，纵坐标为累计奖励。

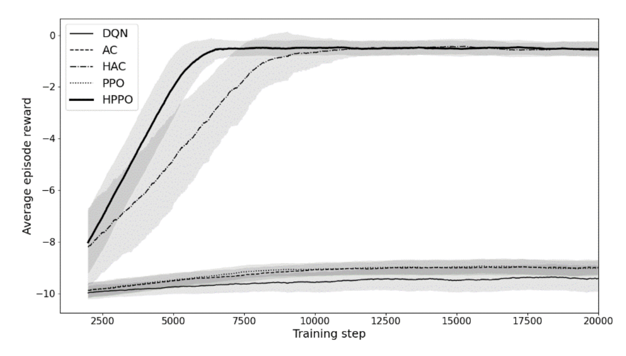


图 8　平均奖励图

Fig. 8　Average reward chart

从结果可以看出，在场景一中，由于实验难度较低，使用分层结构的 HPPO 算法能够在 5000 回合左右实现调头任务，同样具有分层结构的 HAC 算法也在 10000 回合左右的时候实现了调头任务。其他三种算法在 20000 回合都无法达到目标。

使用分层思想改进的 HPPO 算法相比其他算法不仅能够收敛，而且能够以较快的速度进行收敛，这表明了在所有的算法中，HPPO 具有较好的性能。

为了避免偶然性因素，训练好的智能体在进行 50 次仿真后成功掉头的几率以及平均累计回报如表 2 所示。

表 2　不同方法的准确率

Tab. 2　Accuracy of different methods

算法	成功率	平均回报
DQN	0.0007±0.0114	-9.7090±0.3596
PPO	0.0086±0.01252	-9.4957±0.3734
AC	0.0041±0.0146	-8.5556±0.2796
HAC	0.9726±0.0162	-3.0381±2.5415
HPPO	0.9843±0.0124	-2.5021±1.9617

在场景二中，为了增加掉头的难度，减少了仿真车辆在仿真环境的可行驶区域，增加了汽车掉头难度，在该场景下各个算法的表现如图 9 所示。

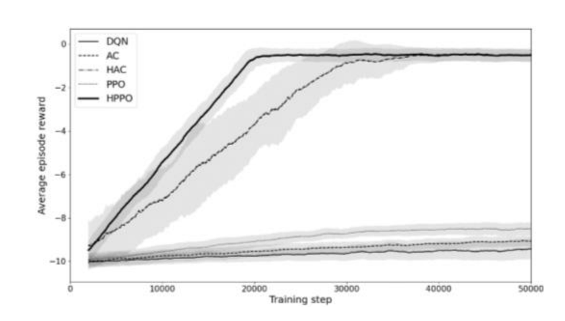


图 9　平均奖励图

Fig. 9　Average reward chart

从结果可以看出，随着训练的不断进行，非分层机制的算法得到的平均奖励在-9 左右，始终无法得到更高的奖励。但相比传统的算法，HPPO 算法能够获得的奖励在不断增加，这说明智能体很好的学习了驾驶技能，也表示 HPPO 算法能够使车辆更加快速安全的完成转弯任务。

同样地，为了避免偶然性因素，训练好的智能体在复杂环境下进行 50 次仿真后成功掉头的几率以及平均累计回报如表 3 所示。

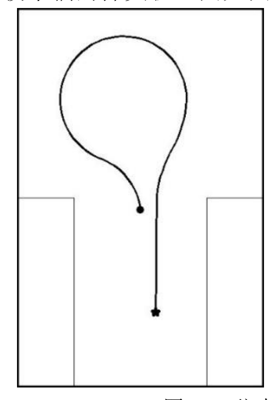
表 3　不同方法的准确率

Tab. 3　Accuracy of different methods

算法	成功率	平均回报
DQN	0.0004±0.0198	-9.7099±0.3596
PPO	0.0181±0.0162	-9.0381±0.5415
AC	0.0021±0.0182	-9.4957±0.3734
HAC	0.9421±0.0152	-3.5556±3.2796
HPPO	0.4248±0.0141	-2.5022±2.9617

4.3　讨论

经过训练后的智能体完全掌握了自动驾驶车辆的掉头任务，且都能在两种掉头场景使用较少的动作成功掉头，自动驾驶车辆的掉头轨迹图如图 10 所示。



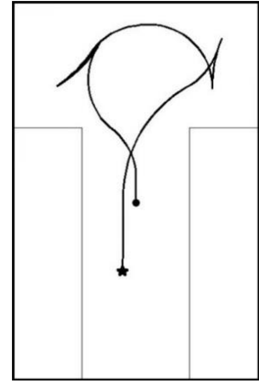


图 10　仿真车辆转弯轨迹图

Fig. 10　Simulation vehicle turning trajectory diagram

如图 10 左图所示，在场景一中，由于仿真车辆的转弯区域较大，所以在训练完成后，车辆仅使用前进转弯动作就完成了掉头任务；如图 10 右图所示，相比左图场景一中的仿真车辆行驶轨迹，场景二的仿真车辆行驶轨迹明显更为复杂。这是因为场景二缩小了仿真车辆的转弯区域，导致转弯难度变大，所以在训练完成后，仿真车辆除了使用前进转弯动作以外，还使用了倒车动作，学习了更多回合才完成了掉头任务。如图 10 所示，仿真车辆的轨迹中实心圆形所在的点表示仿真车辆的掉头起点，实心五角星所在的点代表仿真车辆的掉头终点。

5　结束语

本文针对自动驾驶车辆的掉头问题，首先建立了一个适用于强化学习的马尔可夫决策过程模型，根据实际情形下的

车辆掉头问题，设计了两个场景；然后针对该模型出现的稀疏奖励问题采用分层的思想进行解决，提出了一个分层近端策略优化算法，设计了合理的奖励函数。实验证明，相比于其他传统的强化学习算法，改进后的算法能够在车辆掉头时为车辆设计更安全更快速的掉头策略。

在未来的工作中，考虑自动驾驶汽车其他的小场景问题，旨在适用于更多的场景。

参考文献：

[1] Badue C, Guidolini R, Carneiro R, *et al.* Self-driving cars: A survey [J/OL]. Expert Systems with Applications, 2020, 165: 113816. (2020-08-01) [2021-12-11] DOI: 10. 1016/j. esw a. 2020. 113816.

[2] Qian Lilin, Fu Hao, Li Xiaohui, *et al.* Toward Autonomous Driving in Highway and Urban Environment: HQ3 and IVFC 2017 [M/OL]. IEEE Intelligent Vehicles Symposium (IV) , 2018: 1859. (2018-06-01) [2021-12-11] DOI: 10. 1109/IVS. 20 18. 8500574.

[3] Paden B, Čáp M, Yong S Z, *et al.* A Survey of Motion Planning and Control Techniques for Self-Driving Urban Vehicles [J/OL]. IEEE Trans on Intelligent Vehicles, 2016, 1. (2015-04-25) [2021-12-11] DOI: 10. 1109/TIV. 2016. 25787 06.

[4] Russell S, Norvig P. Artificial intelligence: a modern approach [M]. Prentice Hall, 1995.

[5] Schmidhuber J. Deep learning in neural networks: An overview [J]. Neural Networks, 2015, 61: 85-117.

[6] Sutton R, Barto A. Reinforcement Learning: An Introduction [M]. MIT press, 1998.

[7] Gu Shixiang, Lillicrap T, Sutskever I, *et al.* Continuous Deep Q-Learning with Model-based Acceleration [C]// International conference on machine learning. PMLR, 2016: 2829-2838.

[8] Feinberg V, Wan A, Stoica I, *et al.* Model-Based Value Estimation for Efficient Model-Free Reinforcement Learning [J]. arXiv preprint arXiv: 1803. 00101, 2018.

[9] Mnih V, Kavukcuoglu K, Silver D, *et al.* Playing Atari with Deep Reinforcement Learning [J]. arXiv preprint arXiv: 131 2. 5602, 2013.

[10] Mnih V, Kavukcuoglu K, Silver D, *et al.* Human-level control through deep reinforcement learning [J/OL]. Nature, 2015, 518: 529-533. (2015-02-26) [2021-12-11] DOI: 10. 1038/nature14236.

[11] Van Hasselt H, Guez A, Silver D. Deep Reinforcement Learning with Double Q-learning [C]// Proceedings of the AAAI Conference on artificial Intelligence: 2016.

[12] Schaul T, Quan J, Antonoglou I, *et al.* Prioritized Experience Replay [J]. arXiv preprint arXiv: 1511. 05952, 2015

[13] 刘全，翟建伟，章宗长，等. 深度强化学习综述 [J]. 计算机学报，2018, 41 (01): 1-27. (Liu Quan, Zhai Jianwei, Zhang Zongzhang *et al.* A review of deep reinforcement learning [J]. Journal of Computer Science, 2018, 41 (01): 1-27.)

[14] Chevtchenko S F, Ludermir T B. Combining STDP and binary networks for reinforcement learning from images and sparse rewards [J]. Neural Networks, 2021, 144: 496-506.

[15] Riedmiller M, Hafner R, Lampe T, *et al.* Learning by Playing Solving Sparse Reward Tasks from Scratch [C]// International Conference on Machine Learning. PMLR, 2018: 4344-4353.

[16] Ren H, Ben-tzvi P. Advising reinforcement learning toward scaling agents in continuous control environments with sparse rewards [J]. Engineering Applications of Artificial Intelligence, 2020, 90: 103515.

[17] Jiang N, Jin S, Zhang C. Hierarchical Automatic Curriculum Learning: Converting a Sparse Reward Navigation Task into Dense Reward [J]. Neurocomputing, 2019, 360: 265-278.

[18] Shen C, Chen L, Jia X. A hierarchical deep reinforcement learning framework for intelligent automatic treatment planning of prostate cancer intensity modulated radiation therapy [J]. Physics in Medicine & Biology, 2021, 66 (13): 134002

[19] Frans K, Ho J, Chen X, *et al.* Meta Learning Shared Hierarchies [J]. 2017. arXiv preprint arXiv: 1710. 09767, 2017.

[20] 彭志平，李绍平. 分层强化学习研究进展 [J]. 计算机应用研究，2008, 25 (4): 974-978. (Peng Zhiping, Li Shaoping. Research progress of hierarchical reinforcement learning [J]. Application Research of Computers, 2008, 25 (4): 974-978.)

[21] Li Guofa, Yang Yifan, Li Shen, *et al.* Decision making of autonomous vehicles in lane change scenarios: Deep reinforcement learning approaches with risk awareness [J]. Transportation Research Part C: Emerging Technologies, 2022, 134: 103452.

[22] Peng B, Keskin M F, Kulcsár B, *et al.* Connected autonomous vehicles for improving mixed traffic efficiency in unsignalized intersections with deep reinforcement learning [J]. Communications in Transportation Research, 2021, 1: 100017.

[23] Wang Yunpeng, Zheng Kunxian, Tian Daxin, *et al.* Pre-training with asynchronous supervised learning for reinforcement learning based autonomous driving [J/OL]. Frontiers of Information Technology & Electronic Engineering, 2021, 22: 673-686. (2021-05-01) [2021-12-11] DOI: 10. 1631/FITEE. 1900 637.

[24] Kim J H, Huh J H, Jung S H, *et al.* A Study on an Enhanced Autonomous Driving Simulation Model Based on Reinforcement Learning Using a Collision Prevention Model [J]. Electronics, 2021, 10 (18): 2271.

[25] 张明恒，吕新飞，万星，等. 基于 WGAIL-DDPG (λ) 的车辆自动驾驶决策模型 [J]. 大连理工大学学报，2022, 62 (1): 8. (Zhang Mingheng, Lyu Xinfei, Wan Xing et al, Vehicle autonomous driving decision model based on WGAIL-DDPG (λ) [J]. Journal of Dalian University of Technology, 2022, 62 (1): 8.)

[26] Kendall A, Hawke J, Janz D, *et al.* Learning to drive in a day [C]// 2019 International Conference on Robotics and Automation (ICRA) . IEEE, 2019: 8248-8254.

[27] Rasheed I, Hu Fei, Zhang Lin. Deep reinforcement learning approach for autonomous vehicle systems for maintaining security and safety using LSTM-GAN [J]. Vehicular Communications, 2020, 26: 100266.

[28] Deng Yao, Zhang Tiehua, Lou Guannan, *et al.* Deep learning-based autonomous driving systems: a survey of attacks and defenses [J]. IEEE Transactions on Industrial Informatics, 2021, 17 (12): 7897-7912.